*De novo* transcriptomes built from hundreds of eye tissues reveal hundreds of novel gene isoforms

Vinay S Swamy, Temesgen D Fufa, Robert B Hufnagel, David M McGaughey

# Introduction

       The transcriptome is defined as the set of distinct RNA transcripts expressed in a population of identical cells. During transcription several RNA processing steps modify immature RNA and drive the formation of multiple, distinct gene isoforms. For example the human Gencode release 28 contains 97,713 protein conding transcripts across 20,306 genes. This processing includes alternative promoter usage, alternative splicing, and alternative polyadenylation. The full biological impact of gene isoforms has not been fully elucidated, but multiple studies have shown that gene isoforms can have distinct and critical biological functions [CITE SOMETHINGS].  
       RNA-seq has been used as a powerful tool for complicated task of detecting of novel gene isoforms. Early methods using RNA-seq to detect gene isoforms focused investigating the single? RNA processing mechanism and determining what gene isoforms it was driving. For example, the computational tool rMATS detects novel gene isoforms by analyzing splicing patterns using RNA-seq (1).[MENTION OTHER METHODS]More recently, researchers have adapted methods from genome assembly to use RNA-seq to reconstruct the whole transcriptome of a biological samples, dubbed de novo transcriptome construction (2).  
       De novo transcriptome construction uses short (~100bp) RNA-seq reads to reconstruct full-length mRNA transcripts. This is a particularly useful method because it is independent of any RNA processing mechanism. A major shortcoming of this method is that a large number of samples are required to combat the noisy nature of RNA-seq data, but because of the increasingly inexpensive sequencing, data sets of the necessary size are now available. The most comprehensive de novo transcriptome project to date has been CHESS, which used the GTEx dataset to construct de novo transcriptomes in over 9000 RNA-seq samples from 49 distinct location of the body to create a comprehensive annotation of mRNA transcripts across the human body. (3),(4),(5) However, as the GTEx dataset lacks any ocular tissues, the CHESS database is an incomplete annotation of the human transcriptome.

       A shortcoming of short read based de novo transcriptomes is the inability to evaluate the accuracy of construction methods <- This should be rewritten to something more like “short read de novo transcriptome assemblies precision and sensitivity has not been fully addressed”. Long read sequencing technologies provide a potential solution to this problem as long read sequencing can capture full length transcripts and thus can be used to identify a fuller range of gene isoforms diversity. While long reads have historicaly been considered inaccurate, the new PacBio Sequel II system sequences long reads as accurately as short read based seqencing [CITATION]. We propose that long read based transcriptomes can serve as a ground truth for evaluating short-read base transcriptomes, and in this study we use PacBio long read RNA sequencing to inform the construction of short read transcriptomes. We generated PacBio long read RNA seq data from a stem cell derived retinal pigmented epithelium (RPE) cell line along with matched Illumina short read RNA-seq. Using the two sources of RNA-seq data we design a rigorous de novo transcriptome pipeline that maximizes the conordance between short and long read de novo transcriptomes.

       We further apply this pipeline to construct a pan eye transcriptome using a previously published data set containing 368 368 ocular tissue samples compiled from mining publicly available sequencing data (6).We use this pipeline to build transcriptomes in three major ocular subtissues: The cornea, retina, and the RPE, using RNA-seq data from both adult and fetal tissues to create a high-quality pan-eye transcriptome. In addition to our ocular samples, we used a subset of the GTEx dataset to construct transcriptomes for 49 other locations across the body to facilitate comparisons in transcriptomes across the body.  
       We use our pan eye de novo transcriptome to reveal hundreds of novel gene isoforms as well as several novel genes in the and analyze their potential impact on ocular biology and disease. We provide our de novo transcriptomes as a resource to other researchers through an R package and R-shiny webapp.

# Methods

fig

fig

Figure 0. DNTX workflow

## Generation of PacBio long read RNA sequencing data and Ilumina short read RNA sequencing

## Analysis of Long Read Data

      PacBio hifi reads were processed into full length, non chimeric(FLNC) read using the Pacbio SMRT link software [VERSION]. We next adapted the ENCODE Long Read analysis pipeline: Transcripts were aligned to the human genome using the minimap2(7) with the following parameters. Aligned long reads were processed with TranscriptClean(8) to remove errors due to sequencing. The long read transcriptome was generated with TALON (9)

## Analysis of short read RPE data

      We aligned each sample to the Gencode release 28 (v28) hg38 assembly using the genomic aligner STAR and sorted the resulting BAM files using samtools sort.(10),(11),(12). For each sorted BAM file, we constructed a per sample base transcriptome using StingTie with the Gencode v28 comprehensive annotation as a guiding annotation and default parameters.(10),(3). All sample transcriptomes were merged with the long read transcriptome using gffcompare(13). We define the metric construction accuracy sued to evaluate short read transcriptome construction as the following:

## Construction of tissue specific transcriptomes.

       We identified studies with healthy, unperturbed RNA-seq samples from 52 distinct subtissue regions of the body, downloaded and performed quality control the pertinent sequencing data from the sequence read archive (SRA) using methods from our previous work(6). We constructed a transcriptome for each sample, and merged samples together to create 52 tissue specific base transcriptomes. For each tissue-specific transcriptome, we removed transcripts that had an average expression less than 1 TPM. All tissue specific transcriptomes were merged to form a single unified gtf to ensure transcript identifiers were the same across tissues. We additionally merged all ocular tissue transcriptomes to generate a single pan-eye transcriptome.

## Tissue specific transcriptome quantification

       For each resulting tissue specific transcriptome, we extracted transcript sequences using the tool gffread(13), and used these sequences to build a tissue-specific quantification index using the index mode of the alignment free quantification tool Salmon using the following parameters: .(14) For each sample, we quantified transcript expression using the quant mode of salmon, using a samples respective tissue specific quantification index with the following runtime parameters: –gcBias –seqBias –validateMappings. We similarly quantified all ocular samples using the pan-eye transcriptome We also similarly quantified all samples to the Gencode v28 reference.

## Downstream analysis of Novel transcripts

Analysis of novel exons was done using a custom Rscript [GIVE NAME OF SCRIPT?]. To predict the clinical significance of clinvar variants of uncertain significance, we selected variants from the ClinVar database labeled as “Uncertain Significance”, and used Ensembl’s Variant Effect Predictor tool.

## Identification of protein coding novel transcripts.

       We identified protein coding transcripts in our unified transcriptome using the TransDecoder suite.(2) We extracted transcript sequences using the util script gtf\_genome\_to\_cdna\_fasta.pl and used TransDecoder to find a single best open reading frame from each transcript.We used the agat\_sp\_add\_start\_stop.pl scripts from the AGAT tool to identify start and stop codons for each ORF. Transcripts with no detectable ORF or missing a start or stop codon were marked as noncoding.

## Prediction of novel loci function

Protein coding novel loci were compared to the uniprot protein sequence data base using blastp(15). Blastp results were integrated with hmmer, which identified protein families and domains associated with each novel loci

## Computing Resources

       All computation was performed on the National Institutes of Health cluster compute system Biowulf.

## Code availability.

       To improve reproducibility, we wrote all code used to generate both the data and figures for this paper as Snakemake pipelines.(16) All code used for this projecgt is publicly available in the following github repositories: <https://github.com/vinay-swamy/ocular_transcriptomes_pipeline> - main pipeline,

<https://github.com/vinay-swamy/pacbio_testing> - long read analysis pipeline, <https://github.com/vinay-swamy/ocular_transcriptomes_paper> - figures and table for this paper.[WE NEED TO MOVE THESE TO DAVIDs REPOS]

## Figures and Tables

       All statistical analyses, figures and tables were generated using the R programming language.(17) A full list of packages and versions can be found in supplementary file session\_info.txt. Additionally, all parameters used for commandline tools can be found in supplementary file Snakefile.

# Results

## Long Read Pacbio RNA sequencing guides *de novo* transcriptome construction

In order to determine the accuracy of short read transcriptome construction, we first generated PacBio long read RNA-seq data and Illumina short read RNA-seq data from a stem cell derived RPE cell line. These cell lines were cultured using a highly optimized protocol, and thus should have minimal biological variation. We used this sequencing data to contruct a long read transcriptome and a short read transcriptome. In our long read transcriptome we find XXX distinct transcripts, and in our short read transcriptome YYY distinct transcripts

sum\_counts <- length\_df\_plotting %>%   
 filter(intersection\_case %in% c('stringtie','pacbio','stringtie-pacbio')) %>%   
 mutate(intersection\_case = gsub('stringtie', 'StringTie', intersection\_case),  
 intersection\_case = gsub('pacbio', 'PacBio', intersection\_case),  
 intersection\_case = gsub('StringTie-PacBio', 'PacBio &\nStringTie', intersection\_case)) %>% group\_by(intersection\_case) %>% count()  
violin\_plot <- length\_df\_plotting %>%   
 filter(intersection\_case %in% c('stringtie','pacbio','stringtie-pacbio')) %>%   
 mutate(intersection\_case = gsub('stringtie', 'StringTie', intersection\_case),  
 intersection\_case = gsub('pacbio', 'PacBio', intersection\_case),  
 intersection\_case = gsub('StringTie-PacBio', 'PacBio &\nStringTie', intersection\_case)) %>%   
 ggplot() +  
 scale\_y\_continuous(breaks = c(2000,4000,6000, 8000, 10000)) +  
 geom\_text(data = sum\_counts, aes(x=intersection\_case, y = 11000, label=n), stat= 'identity') +  
 geom\_violin(aes(x=intersection\_case,y=length)) + geom\_hline(yintercept = 2000, color='red') + cowplot::theme\_cowplot() + ylab('Transcript\nLength (bp)') + xlab('') + coord\_flip(ylim = c(0,11500))  
  
tpm\_exp\_plot <- ggplot(data = tpm\_df\_plotting %>% ungroup, aes(x=co, y=acc, color=`Length Interval`)) +   
 geom\_point() +   
 geom\_line()+  
 #xlim(c(-3,26))+  
 #ylim(c(0,1.3))+  
 #geom\_label\_repel(data = labdf\_plotting%>% ungroup, aes(x=co, y=y, label = total\_tx), fill='white', color='black', nudge\_x = -1.5, nudge\_y = .25) +  
 labs(label = 'asf')+  
 ylab('Build Accuracy\n(StringTie ∩ PacBio / StringTie)') +  
 xlab('TPM threshold') +  
 facet\_wrap(~filtering\_type) +  
 cowplot::theme\_cowplot() +  
 scale\_color\_viridis\_d()  
  
violin\_plot/tpm\_exp\_plot +plot\_annotation(tag\_levels = 'A') + plot\_layout(heights = c(1,1.7))

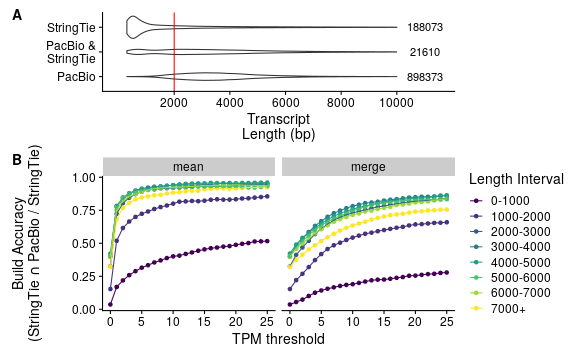


Figure 1. Transcript construction length is substantially larger in the long-read based approach. A) Intersection of transcript lengths between Pacbio (long read) and Stringtie (short read) transcriptomes. The total number of contructed transcripts is given in the text to the right of the violin plot. B) Short read contruction accuracy stratified by transcript length at different TPM based transcript exclusion thresholds.

    In our initial comparison between short and and long read transcriptomes, we see a low transcriptome contruction accuracy (see Methods) of 0.204. When examining the transcript lengths of each build we see that the two methods show very different transcript length distributions, with the short read build comprised mostly of smaller transcripts (Fig 1A) [PUT IN MEDIAN AND MEAN]. As the PacBio data was generated using two libraries for 2000 bp and >3000 bp (need to double check this), we expected to see enrichment for longer transcripts in the pacbio. To assess accuracy relative to transcript length, we group transcripts by length in 1000 bp intervals, and compare accuracy between each group. We found the accuracy significantly improves for transcripts longer than 2000 bp, with construction accuracy 0.389, and 0.095 for transcripts below 2000 bp  
    We experimented with various methods to remove spurious transcripts to improve construction accuracy (bring up ML stuff briefly?), and found that removing transcripts that were expressed below an average TPM threshold was the most effective (Fig1b). We note that this achieves significantly higher accuracy than using the default filtering scheme provided in StringTie, merge [YOU NEED TO EXPLAIN MORE EXPLICITLY WHAT “MEAN” AND “MERGE” ARE). In our downstream pipeline we use an average TPM of 1 (ACROSS WHAT?) as our filtering threshold. This achieves 0.761 accuracy in transcripts longer than 2000BP. This filtering scheme removes a significant number of transcripts from the short read transcriptome, going from XXX to YYY total transcripts.

## A rigorous analysis pipeline finds thousands of novel gene isoforms

df <- sample\_table %>%   
 left\_join(core\_tight) %>%   
 filter(!body\_location %in% c('Body', 'Brain', 'ESC\_Stem.Cell.Line', 'Lens\_Stem.Cell.Line')) %>%  
 group\_by(subtissue) %>%  
 summarise(`Samples` = n(), `Studies` = length(unique(study))) %>%   
 arrange(desc(`Studies`)) %>%   
 inner\_join(tx\_counts %>% select(subtissue, `Transcriptome Count`=filtered)) %>%   
 mutate(subtissue=gsub('\_|\\.',' ', subtissue)) %>%   
 mutate(subtissue = gsub(' Tissue', '', subtissue)) %>%   
 separate(subtissue, into = c('Tissue' , 'Source'), sep = ' ')

## Joining, by = "sample"

## `summarise()` ungroup (override with `.groups` argument)

## Joining, by = "subtissue"

kable(df)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tissue | Source | Samples | Studies | Transcriptome Count |
| Retina | Adult | 105 | 8 | 38961 |
| RPE | Fetal | 49 | 7 | 41641 |
| Cornea | Adult | 43 | 6 | 43897 |
| Retina | Fetal | 89 | 6 | 50584 |
| RPE | Adult | 48 | 4 | 27877 |
| Cornea | Fetal | 6 | 2 | 57712 |

Table 1. Ocular sample dataset overview and transcriptome size. Transcriptome size is defined as the number of unique transcripts expressed in a given tissue type

      We built transcriptomes from 368 published, publicly available ocular tissue RNA-seq samples using an efficient snakemake pipeline (sup fig 1). We include both adult and fetal tissue from cornea, retina, and RPE tissues mined from 29 different studies (Table 1). Our fetal tissues consist of both human fetal tissues and human induced pluripotent stem cell (iPSC) derived tissue, as stem cell derived tissue has been show to closely resemble fetal tissue.[CITATIONS]. To more accurately determine tissue specificity of novel ocular transcripts, we supplemented our ocular data set with 877 samples across 46 body locations from the GTEx project and constructed transcriptomes for each of these body locations. We refer to each distinct body location as a subtissue here after.  
      After initial construction of transcriptomes, we found over two million distinct transcripts across all our samples. After using the filtering methods desribed above, Our final transcriptome across all tissues contains 100244[THIS IS WRONG] distinct transcripts with 88045 previously annotated and 197696 novel transcripts, and incudes 29.868408 megabases of previously unannotated genomicsequence that has not been previously annotated within the Gencode, Ensembl and Refseq annotation databases (10),(18),(19).  
      We split novel transcripts are into two categories: novel isoforms, which are novel variations of known genes, and novel loci, which are previously unreported, entirely novel regions of transcribed sequence. Novel isoforms are further classified by the novelty of its encoded protein: an isoform with novel open reading frame, a novel isoform with a known ORF, and isoforms with no orf as noncoding isoforms. The number of distinct ORFs is significantly less than the number of transcripts, with 41201 previously annotated ORFs and 26593 novel ORFs across all tissues. Across all tissues there is an average of 5585.9 novel isoforms and 1510.81 novel ORFs.

novel\_isoforms\_per\_tissue <- inner\_join(novel\_transcripts\_per\_tissue, novel\_orfs\_per\_tissue) %>%   
 select(-novel\_pc\_count) %>%   
 gather(key = 'transcript\_type', value = 'counts', -body\_location, -color) %>%   
 filter(!body\_location %in% c('Lens\_Stem.Cell.Line', 'ESC\_Stem.Cell.Line')) %>%   
 mutate(body\_location\_pretty=gsub('\_|\\.', ' ', body\_location),  
 body\_location\_pretty=gsub('Tissue','', body\_location\_pretty),  
 body\_location\_pretty=gsub('avg','mean', body\_location\_pretty),  
 body\_location\_pretty=gsub('\\(',' (', body\_location\_pretty),  
 body\_location\_pretty=gsub('Adult','(Adult)', body\_location\_pretty),  
 body\_location\_pretty=gsub('Fetal','(Fetal)', body\_location\_pretty)) %>%   
 mutate(transcript\_type = gsub('\_',' ', transcript\_type),  
 transcript\_type = gsub(' count', '', transcript\_type),  
 transcript\_type = gsub('nc','NC', transcript\_type),  
 transcript\_type = gsub('orf','ORF', transcript\_type),  
 transcript\_type = gsub('novel','Novel', transcript\_type),  
 transcript\_type = gsub('ref','Ref.', transcript\_type))

## Joining, by = c("body\_location", "color")

novel\_loci\_per\_tissue <- novel\_loci\_per\_tissue %>%   
 gather(key = 'transcript\_type', value = 'counts', -body\_location, -color) %>%   
 filter(!body\_location %in% c('Lens\_Stem.Cell.Line', 'ESC\_Stem.Cell.Line')) %>%   
 mutate(transcript\_type = gsub('\_\\w+','', transcript\_type),  
 transcript\_type = gsub('noncoding' ,'Noncoding', transcript\_type),  
 transcript\_type = gsub('pc','Protein\nCoding', transcript\_type)) %>%   
 mutate(body\_location\_pretty=gsub('\_|\\.', ' ', body\_location),  
 body\_location\_pretty=gsub('avg','mean', body\_location\_pretty),  
 body\_location\_pretty=gsub('Tissue','', body\_location\_pretty),  
 body\_location\_pretty=gsub('\\(',' (', body\_location\_pretty),  
 body\_location\_pretty=gsub('Adult','(Adult)', body\_location\_pretty),  
 body\_location\_pretty=gsub('Fetal','(Fetal)', body\_location\_pretty))  
color\_list<- novel\_isoforms\_per\_tissue$color  
names(color\_list) <- novel\_isoforms\_per\_tissue$body\_location\_pretty

loci <- ggplot(data = novel\_loci\_per\_tissue) +  
 geom\_col(aes(x=body\_location\_pretty, y=counts, fill = body\_location\_pretty, alpha = transcript\_type), position = 'dodge') +  
 scale\_fill\_manual(values = color\_list, 'Body\nLocation') +  
 scale\_alpha\_discrete(range=c(.5,1), name = 'Transcript\nType') +  
 #ggtitle('Novel Isoforms Contructed Across the Body')+  
 ylab('Novel\nTranscript\nCount')+  
 xlab('Body Location')+  
 cowplot::theme\_cowplot() +   
 theme(axis.text.x=element\_blank(), panel.background = element\_blank())

## Warning: Using alpha for a discrete variable is not advised.

isotype\_per\_tissue <- ggplot(data = novel\_isoforms\_per\_tissue) +  
 geom\_col(aes(x=transcript\_type, y=counts, fill = body\_location\_pretty), position = 'dodge') +  
 scale\_fill\_manual(values = color\_list) +  
 scale\_alpha\_discrete(range=c(.5,1)) +  
 #ggtitle('Novel Isoforms Contructed Across the Body')+  
 ylab('Noncoding Transcript Count')+  
 xlab('')+  
 cowplot::theme\_cowplot() + theme(legend.position = "none") +   
 theme(axis.text.x=element\_text(angle=45, hjust = 1), panel.background = element\_blank())+  
 facet\_wrap(~body\_location\_pretty, ncol=2)

## Warning: Using alpha for a discrete variable is not advised.

novel\_transcript\_types <- novel\_isoform\_anno\_df %>% filter(label != 'Protein Coding') %>%   
 mutate(group = 'Transcript Type') %>%   
 rename(`Transcript Type` = label) %>%   
 ggplot(data = ., aes(x = group, fill = `Transcript Type`, y=count)) +  
 geom\_bar(position = 'fill',stat = 'identity') +  
 geom\_text(aes(label = count), position = position\_fill(vjust=.5)) +  
 ylab('% of all novel transcripts') +  
 cowplot::theme\_cowplot() +   
 theme(panel.background = element\_blank(), axis.title.x = element\_blank())  
  
novel\_exon\_types <- exon\_type\_by\_transcript\_type %>% group\_by(nv\_type\_rc) %>%   
 summarise(count = n()) %>%   
 mutate(group = 'Novel Exon Type') %>%   
 rename(`Novel Exon Type` = nv\_type\_rc) %>%   
 ggplot(data = ., aes(x = group, fill = `Novel Exon Type`, y=count)) +  
 geom\_bar(position = 'fill',stat = 'identity') +  
 geom\_text(aes(label = count), position = position\_fill(vjust=.5)) +  
 ylab('% of all novel transcripts') +  
 cowplot::theme\_cowplot() +  
 theme(panel.background = element\_blank(), axis.title.x = element\_blank())

## `summarise()` ungroup (override with `.groups` argument)

exon\_type\_tx\_plot <- exon\_type\_by\_transcript\_type %>%   
 mutate(comp\_transcript\_type = gsub('Isoform', 'Isoform,\n', comp\_transcript\_type)) %>%   
 mutate(nv\_type\_rc = gsub('\_',' ', nv\_type\_rc),  
 nv\_type\_rc = gsub('novel', 'Novel', nv\_type\_rc)) %>%   
 group\_by(comp\_transcript\_type, nv\_type\_rc) %>%   
 summarise(count = n()) %>%   
 ggplot(data=., aes(x = comp\_transcript\_type,y=count, fill = nv\_type\_rc)) +  
 geom\_bar(position = 'fill', stat = 'identity')+  
 geom\_text(aes(label = count), position = position\_fill(vjust=.5)) +  
 ylab('Proportion of Transcripts') +  
 cowplot::theme\_cowplot() +  
 theme(axis.text.x=element\_text(angle=45, hjust = 1),   
 panel.background = element\_blank(), axis.title.x = element\_blank()) +  
 scale\_fill\_manual(values = pals::cols25() %>% unname(), name = 'Transcript\nCategory')

## `summarise()` regrouping by 'comp\_transcript\_type' (override with `.groups` argument)

layout <- '  
AB  
AC  
'  
isotype\_per\_tissue + loci + exon\_type\_tx\_plot +plot\_annotation(tag\_levels = 'A') +plot\_layout(design = layout, heights = c(1,5), widths = c(1.5,1), guides = 'collect')

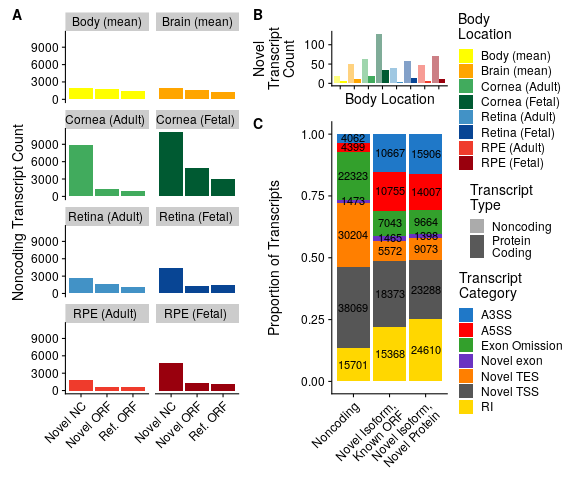


Figure 2. Overview of Novel Isoforms. A. Number of novel gene isoforms, grouped by transcript type. Brain and body represent an average of 13 and 34 distinct subtissues, respectively B. Novel protein coding and noncoding loci. Novel exon composition of novel isoforms, by isoform type labels indicate number of transcripts.

       Novel isoforms occur either due to a novel arrangement of known exons or contain a novel exon (or skipping of known exon?). A novel arrangment of known exons presents as the ommission of one or more exons. Novel exons are either a modification of a known exon to generate a novel modified exon, or an addition of a previously unannotated cryptic exon. For novel exons, we further annotated them with the putative biological process driving their inclusion: alternative splicing for alterative splice sites, retained introns, alternative promoter usage for the inclusion of novel first exons, or alternative polyadenylation for novel last exons. We found that the majority of novel exons with our dataset are novel first and last exons. We see that the majority of A5SS exons lead to novel protein coding isoforms, where as novel TES/TSS more often lead to noncoding isoforms. Retained introns mostly commonly lead to a novel isoform.

## *de novo* transcriptomes reduce overall transcriptome sizes

map\_rate\_diffs <- all\_sample\_mapping\_rate\_difference %>%  
 filter(!body\_location %in% c('Lens\_Stem.Cell.Line', 'ESC\_Stem.Cell.Line')) %>%  
 mutate(body\_location\_pretty=gsub('\_|\\.', ' ', body\_location),  
 body\_location=case\_when(body\_location == 'Body' ~ 'Body(mean)',  
 body\_location == 'Brain'~ 'Brain(mean)' ,  
 TRUE ~ body\_location),  
 body\_location\_pretty=gsub('\_|\\.', ' ', body\_location),  
 body\_location\_pretty=gsub('avg','mean', body\_location\_pretty),  
 body\_location\_pretty=gsub('Tissue','', body\_location\_pretty),  
 body\_location\_pretty=gsub('\\(',' (', body\_location\_pretty),  
 body\_location\_pretty=gsub('Adult','(Adult)', body\_location\_pretty),  
 body\_location\_pretty=gsub('Fetal','(Fetal)', body\_location\_pretty),  
 mapping\_rate\_diff = mapping\_rate\_diff \* -1 ,  
 mapping\_rate\_diff = replace(mapping\_rate\_diff, mapping\_rate\_diff <=0, 0),  
 pt = 'mapping rate reduction')  
  
size\_reduction\_df <- inner\_join(gencode\_tx\_size, tx\_counts) %>%  
 mutate(percent\_txome\_size\_decrease = ( all\_exp-filtered) / all\_exp) %>%  
 inner\_join(subtissue\_to\_bodyloc) %>%  
 filter(!body\_location %in% c('Lens\_Stem.Cell.Line', 'ESC\_Stem.Cell.Line')) %>%  
 mutate(body\_location\_pretty=gsub('\_|\\.', ' ', body\_location),  
 body\_location=case\_when(body\_location == 'Body' ~ 'Body(mean)',  
 body\_location == 'Brain'~ 'Brain(mean)' ,  
 TRUE ~ body\_location)) %>%   
 mutate(body\_location\_pretty=gsub('\_|\\.', ' ', body\_location),  
 body\_location\_pretty=gsub('avg','mean', body\_location\_pretty),  
 body\_location\_pretty=gsub('Tissue','', body\_location\_pretty),  
 body\_location\_pretty=gsub('\\(',' (', body\_location\_pretty),  
 body\_location\_pretty=gsub('Adult','(Adult)', body\_location\_pretty),  
 body\_location\_pretty=gsub('Fetal','(Fetal)', body\_location\_pretty)) %>%   
 group\_by(body\_location\_pretty) %>%  
 summarise(percent\_txome\_size\_decrease = mean(percent\_txome\_size\_decrease)) %>%  
 mutate(pt = '% Base txome removed')

## Joining, by = "subtissue"  
## Joining, by = "subtissue"

## `summarise()` ungroup (override with `.groups` argument)

cl<- novel\_transcripts\_per\_tissue$color  
names(cl) <- novel\_transcripts\_per\_tissue$body\_location  
ggplot(map\_rate\_diffs) +  
 geom\_boxplot(aes(x = pt, y = mapping\_rate\_diff, fill = body\_location\_pretty))+  
 geom\_col( data = size\_reduction\_df, aes(x = pt, y=percent\_txome\_size\_decrease, fill = body\_location\_pretty)) +  
 scale\_fill\_manual(values = color\_list, ) +  
 theme(axis.text.x=element\_text(angle=45, hjust = 1), panel.background = element\_blank()) +  
 facet\_wrap(~body\_location\_pretty, ncol = 2) +  
 ylab('fraction decrease') +  
 theme(legend.position = "none")

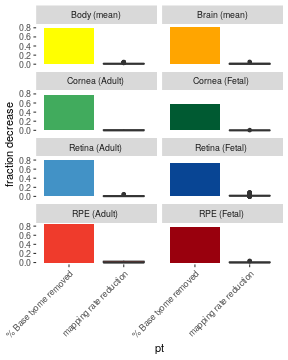


Figure 3. Comparison of Salmon mapping rate decrease vs transcriptome size decrease. Any gain in mapping rate has been set to 0 for simiplicity

      Our transcriptomes removed on average of 79.004 % of the a tissue’s base transcriptome. We define base transcriptome for a tissues as any transcript in the gencode annotation with non zero TPM in at least one sample of a given tissue type. As this was a strikingly large reduction in transcriptome size, we wanted to ensure we were not unduly throwing away data. We quantified transcript expression of our samples using Salmon, quantifiying each sample twice: once using the gencode V27 human transcript annotation, and once using its associated tissue specific transcriptome We found that despite the 79.004 % reduction in number of transcripts between the base gencode and denovo transcriptomes, there is only an average decrease of 0.198 % in mapping rate, indicating that majority of gene expression data is preserved.

## De novo transcriptomes identify of the tissue specific primary isoform expressed in a tissue

    It is useful researchers to know which isoform of a gene is the most relecant to a particular biological system. The APPRIS project identified principal isoforms for humans, based on predicted function and evolutionary conservation, using multiple computational tools. Principal isoforms are considered the most phenotypically relevant isoforms of transcripts. We instead propose a tissue specific primary isoforms, defined as the highest expressed isoforms of a given gene in a given tissue. As primary isoforms are based on biological evidence via RNA-seq, we reason that it is better reflects biology than APPRIS principal isoforms.

primary\_isoforms\_per\_tissue <- primary\_isoforms\_per\_tissue %>%  
 mutate(primary\_isoform\_origin = gsub('dntx','DNTx', primary\_isoform\_origin),  
 primary\_isoform\_origin = gsub('gencode', 'GENCODE', primary\_isoform\_origin),  
 primary\_isoform\_origin = gsub('appris', 'APPRIS', primary\_isoform\_origin)) %>%   
 mutate(primary\_isoform\_origin = factor(primary\_isoform\_origin, levels = c('DNTx', 'GENCODE', 'APPRIS'))) %>%   
 inner\_join(subtissue\_to\_bodyloc) %>%  
 filter(!body\_location %in% c('Lens\_Stem.Cell.Line', 'ESC\_Stem.Cell.Line')) %>%  
 mutate(body\_location\_pretty=gsub('\_|\\.', ' ', body\_location),  
 body\_location\_pretty=case\_when(body\_location == 'Body' ~ 'Body(mean)',  
 body\_location == 'Brain'~ 'Brain(mean)' ,  
 TRUE ~ body\_location)) %>%   
 mutate(body\_location\_pretty=gsub('\_|\\.', ' ', body\_location),  
 body\_location\_pretty=gsub('avg','mean', body\_location\_pretty),  
 body\_location\_pretty=gsub('Tissue','', body\_location\_pretty),  
 body\_location\_pretty=gsub('\\(',' (', body\_location\_pretty),  
 body\_location\_pretty=gsub('Adult','(Adult)', body\_location\_pretty),  
 body\_location\_pretty=gsub('Fetal','(Fetal)', body\_location\_pretty)) %>%   
 group\_by(body\_location\_pretty, primary\_isoform\_origin) %>%  
 summarise(count = mean(count)) %>%   
 arrange()

## Joining, by = "subtissue"

## `summarise()` regrouping by 'body\_location\_pretty' (override with `.groups` argument)

ggplot(data = primary\_isoforms\_per\_tissue, aes(x = body\_location\_pretty, y=count, fill = primary\_isoform\_origin))+  
 geom\_bar(position = 'fill', stat = 'identity')+  
 geom\_text(aes(label = as.integer(count) ), position = position\_fill(vjust=.5)) +  
 cowplot::theme\_cowplot() +  
 theme(axis.text.x=element\_text(angle=45, hjust = 1), panel.background = element\_blank()) +  
 scale\_fill\_brewer(palette = 'Set1', name = 'Origin') + ylab('Ratio') + xlab('')

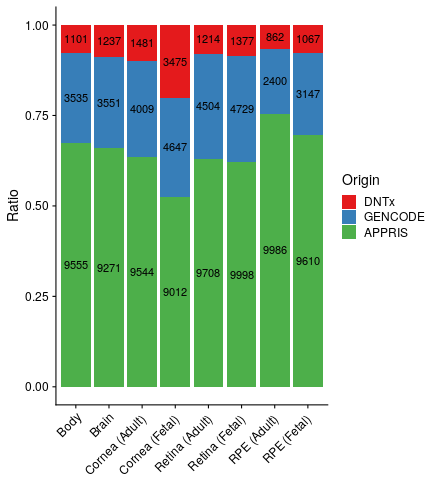


Figure 4. APPPRIS principal isoforms within the de novo transcriptome

   We found that 0.7805 of the APPRIS principal isoforms are also identified as the primary isoform in at least one tissue type. Within each tissue type, on average 0.670765989163849 % of are an appris primary isoform, 0.245203919195482 of transcripts were a previously annotated gencode transcript, 14242.8461538462 [THIS IS WRONG] % of principal isoforms are novel isoforms.

## Novel Isoforms in Ocular tissues

source('scripts/R\_raincloud.R')  
plot\_list <- novel\_eye\_tx\_by\_tissue[!names(novel\_eye\_tx\_by\_tissue) %in% c('Lens\_Stem.Cell.Line', 'ESC\_Stem.Cell.Line') ]  
all\_tx <- tibble(transcript\_id = reduce(plot\_list, union))  
plot\_df <- bind\_cols(all\_tx, lapply(plot\_list, function(x) all\_tx$transcript\_id%in% x))  
FIG\_FONT\_SIZE=12  
  
colnames(plot\_df) <- gsub('\_|\\.', ' ', colnames(plot\_df) )   
colnames(plot\_df) <- gsub('Adult', '(Adult)', colnames(plot\_df) )  
colnames(plot\_df) <- gsub('Fetal', '(Fetal)', colnames(plot\_df) )  
colnames(plot\_df) <- gsub('Tissue', '', colnames(plot\_df) )  
  
  
us <- ComplexUpset::upset(data = plot\_df,   
 intersect = colnames(plot\_df)[-1],   
 min\_size= 500,   
 set\_sizes = F,  
 #height\_ratio = .8,   
 #width\_ratio = .1,  
 wrap = T,  
 stripes = c('white','white'),  
 base\_annotations = list(  
 'Intersection size' = intersection\_size(text\_colors = c(on\_background='red', on\_bar='red'))  
 ),  
 themes=upset\_modify\_themes(  
 list(  
 "intersections\_matrix"= theme(text = element\_text(size=FIG\_FONT\_SIZE) ),  
 "Intersection size"= theme(text = element\_text(size=FIG\_FONT\_SIZE),  
 panel.grid.major = element\_blank(),   
 panel.grid.minor = element\_blank()),  
 "overall\_sizes"= theme(axis.text.x=element\_text(angle=45),text = element\_text(size=FIG\_FONT\_SIZE) ),  
 "default"= theme(text = element\_text(size=FIG\_FONT\_SIZE))   
 )  
 )  
   
)  
  
plotting\_piu\_df <- piu\_df %>% mutate(new\_stage=ifelse(stage == 'adult', 'Adult', 'Fetal'),   
 subtissue=paste0(tissue, '\_', new\_stage, '.Tissue')) %>%   
 mutate(subtissue=gsub('\_|\\.', ' ', subtissue),  
 subtissue=gsub('avg','mean', subtissue),  
 subtissue=gsub('Tissue','', subtissue),  
 subtissue=gsub('\\(',' (', subtissue),  
 subtissue=gsub('Adult','(Adult)', subtissue),  
 subtissue=gsub('Fetal','(Fetal)', subtissue))  
  
piu <- ggplot(data=plotting\_piu\_df, aes(x=stage, y=piu, fill=subtissue, color=subtissue)) +   
 geom\_flat\_violin(position = position\_nudge(x = .2, y = 0),adjust = 2)+  
 geom\_point(position = position\_jitter(width = .15), size = .25, alpha=.2) +   
 geom\_boxplot(outlier.shape = NA, alpha = 1, width = .1, colour = "BLACK")+   
 scale\_fill\_manual(values = color\_list) +  
 scale\_color\_manual(values = color\_list) +  
 ylab('piu')+xlab('tissue')+coord\_flip()+   
 cowplot::theme\_cowplot()+ facet\_wrap(~tissue, nrow = 3) +  
 theme(legend.position = "none")  
  
us + piu + plot\_layout(widths = c(2,1)) + plot\_annotation(tag\_levels = 'A') & theme(plot.tag = element\_text(size= FIG\_FONT\_SIZE), text = element\_text(size = FIG\_FONT\_SIZE))

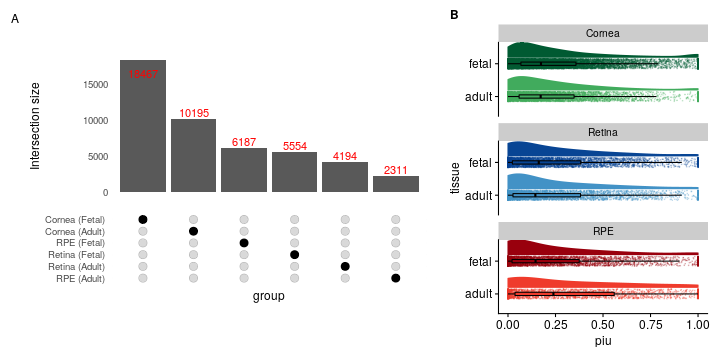


Figure 5. Analysis of novel isoform characteristics A. Set intersection of novel isoforms in ocular transcriptomes. B Boxplots of PIU overlaid over PIU data points with estimated distribution of data set above each boxplot

      Next, we analyzed the novel isoforms within our ocular transcriptomes. We compared the overlap in constructed novel isoforms across ocular tissues and found that XXX % of novel isoforms are specific to a singular ocular subtissue (Fig 3A), but there are significant transcripts clusters for pan-eye and fetal-like tissues as well. For each novel isoform we then calculated percent isoform usage (PIU), or the fraction of total gene expression a transcript contributed to its parent gene. We found that on average novel isoforms contribute to 15% of their parent gene’s expression, although we see multiple novel isoforms that are the primary isoform for their parent gene.

## Novel Loci in the ocular transcriptome

We determined the longest ORF’s for each novel loci, and for each transcript with a valid ORF, predicted the function the encoded protein using hmmer searching within the pfam databse, and blastp searching within the swissprot database. We chose to hmmer results that had a sequence match e-value < 1e-4, and blastp results with e-value < 1e-15 (Sup Table). We found XXX distinct potentially novel coding loci from blast and hmmer combined. We highlight one detected in fetal RPE

library(patchwork)  
  
cov <- ggplot(cov\_df, aes(x=coord,y=cov, group = sample))+  
 geom\_line( alpha = .3)+  
 geom\_area(alpha = .3)+  
 #ggtitle(paste(txid, s\_st))+  
 theme(axis.title.x = element\_blank(), panel.background = element\_blank(),   
 axis.text.x = element\_blank(), axis.ticks.x = element\_blank())  
pt <- ifelse(gene\_model\_df$strand[1] == '+',62, 60)  
model <- ggplot(gene\_model\_df)+  
 geom\_rect(aes(xmin=xmin, xmax=xmax, ymin=ymin, ymax=ymax))+  
 geom\_rect(xmin=min(gene\_model\_df$xmin), xmax=max(gene\_model\_df$xmax), ymin=-.1, ymax=.1)+  
 geom\_point(data=point\_df, aes(x = x, y=y), shape= pt, size = 5)+  
 xlab('genome') +  
 ylab(element\_blank())+  
 theme(panel.background = element\_blank(), axis.text.y = element\_blank(), axis.ticks.y = element\_blank())  
layout <- '  
AA  
AA  
AA  
AA  
AA  
AA  
BB  
CC  
CC  
'  
mss <- as\_ggplot(rasterGrob(readPNG('clean\_data/plots/msa\_ex.png'), interpolate = T))  
cov/model/mss + plot\_layout(design = layout) +plot\_annotation(tag\_levels = 'A')

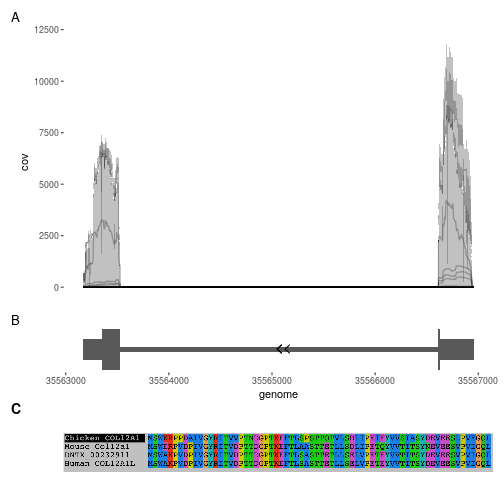


Figure 6. DNTX000dddd, potentially protein coding novel isoform in Fetal RPE. A) RNA-seq coverage of exons across 49 samples. B transcript model of DNTX000ddd. thick regions represtn coding region. C. Multiple sequence alignment of DNTX000ddd and closely related proteins.

Placeholder figure, still trying to find the right transcript to look at here.

## De novo transcriptome allow for a more precise variant prioritaizaion.

The prediction of variants’s biological impact is a fundamental step in variant prioritzation, especially in the diagnosis of genetic diseases or in interpreting GWAS results. We use our Ensembl’s Variant Effect Predictor combined with our tissue transcriptomes to generate a tissue specific prediction of variant impact for variants of unknown significance(VUS) the clinvar database. We grouped different predicted consequences into a High, moderate and low impacts, and provide a full mapping of biological consequence to impact in our supplementary data.

n=10  
n=10  
vep\_exp\_all\_eye <- filter(dntx\_vep\_impact\_matrix, allele\_id %in% clinvar\_vus\_eye$allele\_id) %>%  
 filter(rowSums(.[,eye\_tissues] == 3) >=1, rowSums(.[,body\_tissues] !=3) ==length(body\_tissues))  
vep\_no\_eye <- filter(dntx\_vep\_impact\_matrix, allele\_id %in% clinvar\_vus\_eye$allele\_id) %>%   
 filter(rowSums(.[,eye\_tissues] != 3) == length(eye\_tissues),three\_count >4 )   
impact\_diff\_changes <- impact\_diff %>% filter(min\_diff!=0 | max\_diff!=0)  
eye\_impact\_increase <- impact\_diff\_changes %>% filter(max\_diff>0, allele\_id%in% clinvar\_vus\_eye$allele\_id)  
vep\_impact\_increase <- filter(dntx\_vep\_impact\_matrix, allele\_id %in% eye\_impact\_increase$allele\_id)  
  
  
vep\_example <- bind\_rows( vep\_exp\_all\_eye %>% arrange(desc(var))%>%   
 #.[1:n,] %>%   
 select(allele\_id, eye\_tissues, everything()),  
 vep\_no\_eye %>% arrange(desc(var)) %>%   
 #.[1:n,] %>%   
 select(allele\_id, eye\_tissues, everything()) ,  
 vep\_impact\_increase  
 )

## Note: Using an external vector in selections is ambiguous.  
## ℹ Use `all\_of(eye\_tissues)` instead of `eye\_tissues` to silence this message.  
## ℹ See <https://tidyselect.r-lib.org/reference/faq-external-vector.html>.  
## This message is displayed once per session.

clinvar\_example <- filter(clinvar\_vus\_eye, allele\_id %in% vep\_example$allele\_id) %>% select(allele\_id, contains('pheno'))  
#vep\_example <- filter(vep\_impact\_matrix, allele\_id %in% clinvar\_vus\_eye$allele\_id) %>% arrange(desc(var))  
gencode\_vep\_by\_tissue\_simple <- gencode\_vep\_by\_tissue %>%   
 mutate(subtissue = 'gencode', max\_impact\_avg\_exp=1, max\_impact\_med\_exp=1) %>%   
 distinct()  
merge\_bt <- bind\_rows(dntx\_vep\_by\_tissue %>% mutate(build= 'dntx'),  
 gencode\_vep\_by\_tissue\_simple%>% mutate(build= 'gencode'))  
plot\_df <- merge\_bt %>%   
 filter(allele\_id %in% vep\_example$allele\_id) %>%  
 mutate(avg\_exp = log2(max\_impact\_avg\_exp+1),   
 med\_exp = log2(max\_impact\_med\_exp + 1),  
 avg\_exp = replace(avg\_exp , avg\_exp >5, 5),  
 allele\_id = as.character(allele\_id),   
 `Variant Impact` = case\_when(max\_impact == 1 ~'Low',  
 max\_impact == 2 ~ 'Moderate',   
 max\_impact == 3 ~ 'High') %>% factor(levels=c('Low', 'Moderate', 'High')),  
 subtissue = factor(subtissue, levels = c('gencode',eye\_tissues, body\_tissues))  
 ) %>%   
 rename(`log2(TPM + 1)` = avg\_exp)  
plot\_df[is.na(plot\_df)] <- 0  
cl <- c('Low' = 'lightblue', 'Moderate' = 'lightgreen', 'High' = 'red')  
shp <- c('dntx'=16, 'gencode'=15)  
ggplot(plot\_df, aes(x =allele\_id, y=subtissue))+  
 geom\_point(aes(size = `log2(TPM + 1)`, color = `Variant Impact`, shape=build))+  
 scale\_color\_manual(values =cl )+  
 scale\_shape\_manual(values = shp)+  
 geom\_hline(yintercept = 1.5)+  
 ggtitle('Tissue Specific Variant Prioritization') +  
 theme(axis.text.x.bottom = element\_text(angle = 45, hjust = 1), panel.background = element\_blank())

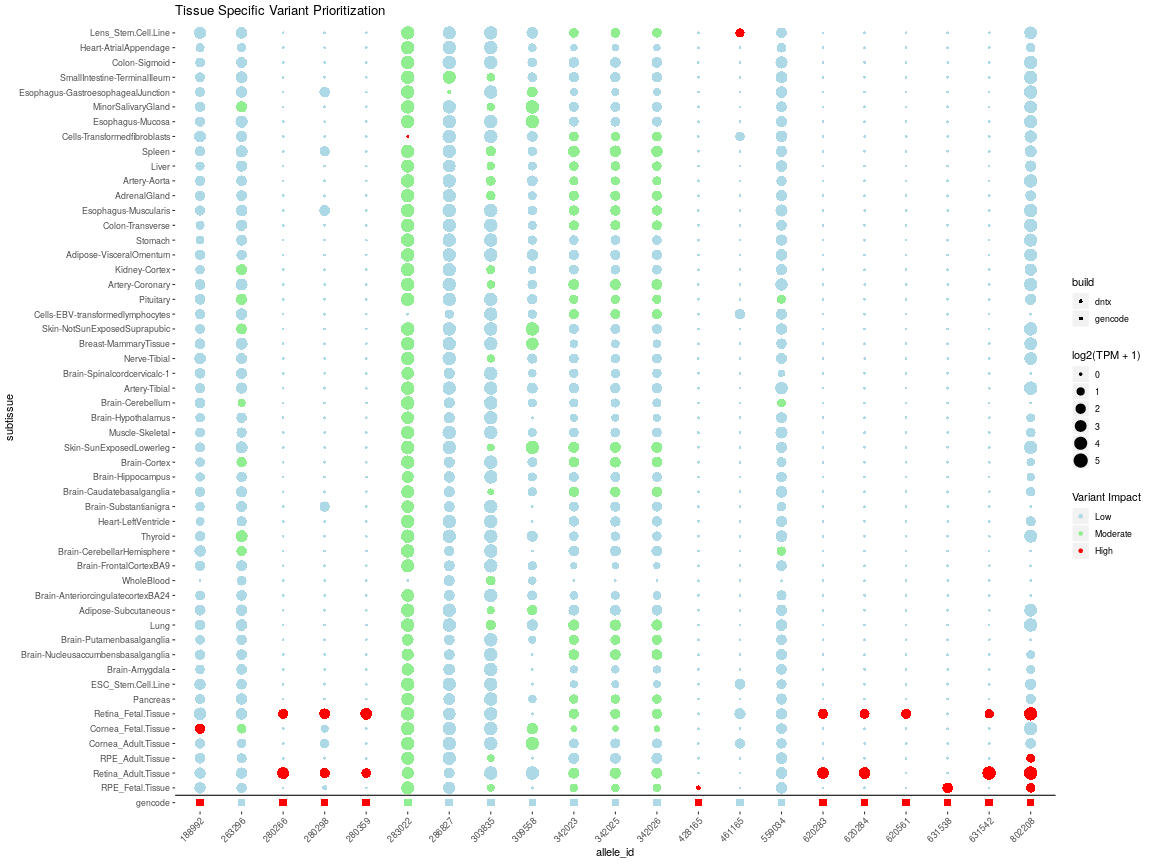


Figure 7. Tissue specific variant prioritization of variants associated with ocular disease. Size corresponds to average TPM and color to impact. A full mapping of variant descriptors to impact can be found in supplemental data. DROP LENS? Collapse GTEx into brain, not brain?? Clean up formatting (underscores and periods).

We compared out tissue specific variant effect predictions to variant effect predictions using the gencode annotation. We found that out 226824 total variants examined,the highest priority of 117556 decreased in at least one tissue, and 1075 increased in at least one tissue. We see a larger decrease in predicted effects as the transcript which previously lead to an effect is not expressed in a respective tissue. We next selected variants that were associated with ocular disease phenotypes, to see if there were any interesting changes. Variant 280266(what ID is this??), , associated retinitis pigmentosa, while considered high impact in gencode as well, its only high impact in adult retina, so more likely to be causative. Variant 843125 associated with retinal pigmentary distrophy, as a high gencode impact, and is high impact in other tissues, low impact and not expressed in any ocular tissue, so likely not disease causing.

## A companion visualization tool enables easy use of *de novo* transcriptomes

piu <- as\_ggplot(rasterGrob(readPNG('clean\_data/plots/piu\_shiny\_app.png'), interpolate = T))  
frac\_det <- as\_ggplot(rasterGrob(readPNG('clean\_data/plots/frac\_sample\_det.png'), interpolate = T))  
gene\_body <- as\_ggplot(rasterGrob(readPNG('clean\_data/plots/gene\_bodies\_shiny\_app.png'), interpolate = T))  
layout <- '  
AABB  
CCCC  
'  
piu + frac\_det + gene\_body + plot\_layout(design = layout)+ plot\_annotation(tag\_levels = 'A')

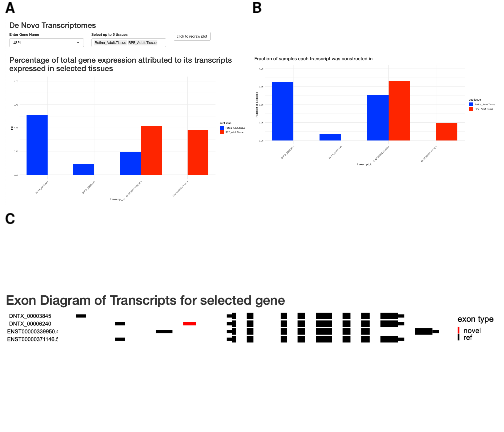


Figure 8. Screenshots from dynamic *de novo* transcriptome visualization tool. A). PIU bar plot for selected gene and tissue. B). Exon level diagram of transcript body Thicklines represent coding region of transcript. novel exons colored in red. Tooltip contains genomic location and phylop score C) Bargraph of fraction of samples within dataset each transcript was consructed in by tissue.

       To make our results easily accessible, we designed a webapp for visualizing and accessing our *de novo* transcriptomes. Users start by selecting gene or search for a gene by genomic location, and selecting up to 5 tissues to visualize transcript expression in. For each tissue we show the PIU for each transcript associated with a gene. We show the exon-intron structure of each transcript and mousing over exons show genomic location overlapping SNPs, and phylogentic conservation score. We additionally show a barplot of the fraction of samples in each tissue each transcript was constructed in. Users can also download the *de novo* transcriptomes for selected tissues in GTF and fasta format. Short mention of how DNTx is in eyeIntegration now.

## A easy-to-use pipeline to enable future research

Finally, we package all tools used for out transcriptome pipeline within a portable docker container with a stand-alone run script. This pipeline allows other researchers to run their own samples, and generate figures and annotations similar to what is shown here. (havent worked it out entirely, but snakemake can automatically use a singulaity container , so if I provide a singularity image anyone can use it)

# Discussion

Motivated by the lack of a comprehensive pan eye transcriptome we created the first comprehensive set of ocular transcriptomes. By using long read RNA-sequencing data to calibrate our short-read construction pipeline, we increase confidence we created real, biologically relevant transcriptomes. We found that concordance between long and short read based transcriptome is directly related to transcript expression and transcript length. We saw a clear inability within this PacBio data to accurately detect transcripts shorter than 2000Bp (wait we do? Is this because they are fake? How do we distinguish? Did you show that PacBio is missing a disproportionate amount of gencode short tx??????). As many of the transcripts constructed using short reads are below this threshold, long read sequencing data enriched for smaller transcript sizes would provide greater insight in future studies. Even when accounting for transcript length, we achieved a contruction accuracy of 0.389. This was surprisingly low. It is difficult to say whether this low concordance it due to a high degree of transcriptional noise or errors in the short read transcriptome construction. We see many more transcripts in the long read data than the short read data which may hint to transcriptional noise; obtaining long read data from more samples and seeing the concordance across multiple samples would help address this hypothesis.

We used a large dataset compiled from published RNA-seq data to build our ocular transcriptomes, an approach which has several key advantages. First, our large sample size allows us to combat the noisy nature of RNA-seq data. More importantly, because we only keep transcripts detected across multiple samples from multiple studies with multiple types of sample preparation, we can be more confident that our transcriptomes accurately reflect the biology of its originating subtissue and are not a technical artifact due to preparation of the samples or the transcriptome construction algorithmn.

We show that across all tissue types, the number of constructed transcripts is dramatically less than the number of transcripts present in the Gencode reference annotation (Table 1; Figure 1a) DO SOME LITERATURE SEARCH TO SEE IF THIS IS MENTIONED ANYWHERE. GTEx papers?). Despite the large reduction in number of transcripts in the annotation, we see little loss in gene expression data. Given this and that we recover XXX% of gencode annotation across all tissues,?? indicates that many of the transcripts in the gencode annotation are not biologically relevant, and that the one size fits all strategy of having a single comprhensive annotation may not be the by the right one. This reduction in total transcriptome sizes is also useful for other researchers as it creates a smaller search space to explore, ie it is easier to compare 30,000 transcripts than 300,000 In almost all tissues we see more noncoding novel isoforms than novel coding. However as all/most of our data was prepared with polyA selection, we likely are not capturing canonical lincRNA/ncRNA which do not have polyA tails. These transcripts fall under the processed\_transcript category by the Gencode annotation which is a very vague category so it is difficult to predict their function. Many of these have alternative FEs so might be under a different regulatory program. Though we identified many novel ORFs it is difficult to say whether these are actually protein coding. We see that many of the novel ORF’s are novel because of retained introns. These retained intron transcirpt may simply be unspliced mRNA. Studies have shown that some neuronal cell types stockpile unspliced transcripts for generating rapid changes in gene expression, and so we may be capturing a simiar phenomenon here. However we are confident that we are capturing true novel protein isoforms because XYZ. We find a lot, but no where near the upper limit suggested by (20)

We observed that in the set of novel exons within our transcriptome, the vast majority are novel first and last exons [NUMBERS]. It is difficult to directly tell what the biological relevance of these might be. There are multiple studies examining the roles of alternative first and last exons, with first exons arising because of alternative promoter usage, and last exons by alternative polyadenylation. (21),(22) These studies have highlighted the distinct biological role these phenomena play. However, other studies have shown that many of these differences in start sites are largely nonfunctional and lack biological significance.(23). Previous research devoted to studying splicing vastly outnumbers other research on other RNA processing types, yet we show that isoforms generated by these mechanisms are equally abundant, and thus merit more study. Something about the reliability of determining novel 5’ ends and 3’ ends from RNA-seq/de novo transcriptomes.

In our ocular transcriptomes, we see that novel isoforms are largely subtissue-specific. This matches previously reported findings about the tissue specificity of rare exons and first and last exons. We also find that on average novel isoforms represent about 15% of their parent. However, it is difficult to say if these are non-functional. Its is entirely possible that these isoforms are from low population cell types. This especially makes sense in the retina, bipolar/amacrine cell types comprise ~15% of the total cell population, and that within these cell types there are even smaller sub cell type populations. However, we note that transcripts only expressed in rare cell types are likely to be missed by our pipeline, as we used for a conservative expression threshold for the filtering step within our pipeline

We want to make our transcriptomes easily accessible to other researchers, so we designed a webapp to visualize our transcriptomes and access tissue-specific annotation files. We wanted to provide as much information to the user so that they can draw their own conclusions about the significance of potential novel exon, and so we provide the gene model with novel exon, coding and noncoding regions marked, along with the PIU for each transcript constructed within a gene. We also provide the fraction of samples within a given subtissue type a sample was detected in, to provide a further level of evidence to the validity of constructed transcripts.

utility of de novo transcriptomes - primary isoforms, variant effect prediction, somthing about precision medicine.

novel loci found some, some are interseting, some are not.

# Notes

* Correlation between transcript length and expression?
* bench marking transdecoder on gencode txome.
* comparison of cage to non cage

# References

1. Shen,S., Park,J.W., Lu,Z.-x., Lin,L., Henry,M.D., Wu,Y.N., Zhou,Q. and Xing,Y. (2014) rMATS: Robust and flexible detection of differential alternative splicing from replicate RNA-Seq data. *Proceedings of the National Academy of Sciences*, **111**, E5593–E5601.

2. Haas,B.J., Papanicolaou,A., Yassour,M., Grabherr,M., Blood,P.D., Bowden,J., Couger,M.B., Eccles,D., Li,B. and Lieber,M. *et al.* (2013) De novo transcript sequence reconstruction from RNA-Seq: Reference generation and analysis with Trinity. *Nature protocols*, **8**.

3. Pertea,M., Pertea,G.M., Antonescu,C.M., Chang,T.-C., Mendell,J.T. and Salzberg,S.L. (2015) StringTie enables improved reconstruction of a transcriptome from RNA-seq reads. *Nature Biotechnology*, **33**, 290–295.

4. GTEx Consortium, Laboratory, Data Analysis &Coordinating Center (LDACC)—Analysis Working Group, Statistical Methods groups—Analysis Working Group, Enhancing GTEx (eGTEx) groups, NIH Common Fund, NIH/NCI, NIH/NHGRI, NIH/NIMH, NIH/NIDA and Biospecimen Collection Source Site—NDRI *et al.* (2017) Genetic effects on gene expression across human tissues. *Nature*, **550**, 204–213.

5. Pertea,M., Shumate,A., Pertea,G., Varabyou,A., Breitwieser,F.P., Chang,Y.-C., Madugundu,A.K., Pandey,A. and Salzberg,S.L. (2018) CHESS: A new human gene catalog curated from thousands of large-scale RNA sequencing experiments reveals extensive transcriptional noise. *Genome Biology*, **19**, 208.

6. Swamy,V. and McGaughey,D. (2019) Eye in a Disk: eyeIntegration Human Pan-Eye and Body Transcriptome Database Version 1.0. *Investigative Ophthalmology & Visual Science*, **60**, 3236–3246.

7. Li,H. (2018) Minimap2: Pairwise alignment for nucleotide sequences. *Bioinformatics (Oxford, England)*, **34**, 3094–3100.

8. Wyman,D. and Mortazavi,A. (2019) TranscriptClean: Variant-aware correction of indels, mismatches and splice junctions in long-read transcripts. *Bioinformatics*, **35**, 340–342.

9. Wyman,D., Balderrama-Gutierrez,G., Reese,F., Jiang,S., Rahmanian,S., Forner,S., Matheos,D., Zeng,W., Williams,B. and Trout,D. *et al.* (2020) A technology-agnostic long-read analysis pipeline for transcriptome discovery and quantification. *bioRxiv*, [10.1101/672931](https://doi.org/10.1101/672931).

10. Frankish,A., Diekhans,M., Ferreira,A.-M., Johnson,R., Jungreis,I., Loveland,J., Mudge,J.M., Sisu,C., Wright,J. and Armstrong,J. *et al.* (2019) GENCODE reference annotation for the human and mouse genomes. *Nucleic acids research*, **47**, D766–D773.

11. Dobin,A., Davis,C.A., Schlesinger,F., Drenkow,J., Zaleski,C., Jha,S., Batut,P., Chaisson,M. and Gingeras,T.R. (2013) STAR: Ultrafast universal RNA-seq aligner. *Bioinformatics (Oxford, England)*, **29**, 15–21.

12. Li,H., Handsaker,B., Wysoker,A., Fennell,T., Ruan,J., Homer,N., Marth,G., Abecasis,G., Durbin,R. and 1000 Genome Project Data Processing Subgroup (2009) The Sequence Alignment/Map format and SAMtools. *Bioinformatics (Oxford, England)*, **25**, 2078–2079.

13. Pertea,G. and Pertea,M. (2020) GFF Utilities: GffRead and GffCompare. *F1000Research*, **9**, 304.

14. Patro,R., Duggal,G., Love,M.I., Irizarry,R.A. and Kingsford,C. (2017) Salmon provides fast and bias-aware quantification of transcript expression. *Nature methods*, **14**, 417–419.

15. Altschul,S.F., Gish,W., Miller,W., Myers,E.W. and Lipman,D.J. (1990) Basic local alignment search tool. *Journal of Molecular Biology*, **215**, 403–410.

16. Köster,J. and Rahmann,S. (2012) Snakemake—a scalable bioinformatics workflow engine. *Bioinformatics*, **28**, 2520–2522.

17. R Core Team (2019) R: A Language and Environment for Statistical Computing R Foundation for Statistical Computing, Vienna, Austria.

18. Zerbino,D.R., Achuthan,P., Akanni,W., Amode,M.R., Barrell,D., Bhai,J., Billis,K., Cummins,C., Gall,A. and Girón,C.G. *et al.* (2018) Ensembl 2018. *Nucleic Acids Research*, **46**, D754–D761.

19. O’Leary,N.A., Wright,M.W., Brister,J.R., Ciufo,S., Haddad,D., McVeigh,R., Rajput,B., Robbertse,B., Smith-White,B. and Ako-Adjei,D. *et al.* (2016) Reference sequence (RefSeq) database at NCBI: Current status, taxonomic expansion, and functional annotation. *Nucleic Acids Research*, **44**, D733–745.

20. Ponomarenko,E.A., Poverennaya,E.V., Ilgisonis,E.V., Pyatnitskiy,M.A., Kopylov,A.T., Zgoda,V.G., Lisitsa,A.V. and Archakov,A.I. (2016) The Size of the Human Proteome: The Width and Depth. *International Journal of Analytical Chemistry*, **2016**.

21. Demircioğlu,D., Cukuroglu,E., Kindermans,M., Nandi,T., Calabrese,C., Fonseca,N.A., Kahles,A., Lehmann,K.-V., Stegle,O. and Brazma,A. *et al.* (2019) A Pan-cancer Transcriptome Analysis Reveals Pervasive Regulation through Alternative Promoters. *Cell*, **178**, 1465–1477.e17.

22. Mitra,M., Johnson,E.L., Swamy,V.S., Nersesian,L.E., Corney,D.C., Robinson,D.G., Taylor,D.G., Ambrus,A.M., Jelinek,D. and Wang,W. *et al.* (2018) Alternative polyadenylation factors link cell cycle to migration. *Genome Biology*, **19**, 176.

23. Xu,C., Park,J.-K. and Zhang,J. (2019) Evidence that alternative transcriptional initiation is largely nonadaptive. *PLOS Biology*, **17**, e3000197.